**Programación de Computadoras II**

**Tarea: 11**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

En relación con la Tarea 10 donde se implemento una plataforma de Dibujo parecida a Paint (de Microsoft), hacer:

1.- Agregar un conjunto de herramientas que permita borrar una figura,

2.- Modificar la clase de figuras para que sea comparable y

3.- Escribir los métodos necesario para saber si cuando estoy en el centro de una figura cual de la lista es

**Programación de Computadoras II**

**Tarea: 10**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Implementar código Java que permita implementar una plataforma de Dibujo parecida a Paint (de Microsoft). Dicha Plataforma deberá tener manera de dibujar líneas, cuadros, rectángulos, círculos y elipses.

Entregar:

1.- El código Java de cada una de las figuras implementadas y

2.- La demostración de la ejecución del código (en computadora a la hora clase).

**Programación de Computadoras II**

**Tarea: 9**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Implementar y hacer funcionar el código correspondiente para la clase Dibuja. Escribir el código necesario para implementar los botones, los radio button de la clase selección y el Lienzo.

Mostrar solamente la evidencia de que el código corre.

**Programación de Computadoras II**

**Tarea: 8**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Para el caso del código Java que modelo de la escuela se quiere modificar el método Lista\_Alumnos\_por\_Materia() tal que produzca una salida como la que se muestra a continuación. El cambio en la salida es agregar el nombre del profesor con su clave y nombre completo

Profesor : P23498dasd Ramirez Calderon Romario

m01 Matemáticas I

Matemáticas I

1.- 8sdsdf8 Juárez Mena Manuel

2.- 02392d Medina Alvarez Carla

3.- 9234jja Rico Gomez Maria Luisa

4.- 8234jf Arce Salas Luis Manuel

Profesor : Materia sin Profesor

Matemáticas II

Profesor : Plcsdff Ricon Lopez Honorio

m03 Física I

Física I

1.- 122kka Pérez López Juan

2.- 8sdsdf8 Juárez Mena Manuel

3.- 8234jf Arce Salas Luis Manuel

**Programación de Computadoras II**

**Tarea: 7**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Normalmente las Facturas agregan en la información la fecha en la que fue expedida. Crear el objeto Fecha que guarde información referente al día, mes y año e incorporarla al objeto Factura. El diagrama UML de fecha es:

|  |
| --- |
| **Fecha** |
| - dia: int - mes : int- Anio : int |
| <<constructor>> Fecha(unDia: int, unMes : int, unAnio:int)+ formato() : String regresa la fecha en formato 01 de enero de 2013 |

Escribir el código correspondiente para llevar a cabo esta implementación.

**Programación de Computadoras II**

**Tarea: 6**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Para el objeto Fracción

class Fraccion{

 private int numerador, denominador;

 Fraccion(int unNum, int unDen){

 int m;

 m = mcd(unNum, unDen);

 numerador = unNum/m;

 denominador = unDen/m;

 }

 String imprime(){

 String aux = "";

 if(numerador/denominador >=0)

 aux += Math.abs(numerador) +"/" +Math.abs(denominador);

 else

 aux += "-" + Math.abs(numerador) +"/" +Math.abs(denominador);

 return aux;

 }

 private int mcd(int a, int b) {

 if(b == 0) return a;

 else return mcd(b, a%b);

 }

 ...

 ...

}

Cuando se ejecuta

static public void main(String args[]){

 Fraccion a = new Fraccion(63,0);

 System.out.println(a.imprime());

}

Se genera el siguiente mensaje de error, note que este es generado en la linea correspondiente al método imprime.

Exception in thread "main" java.lang.ArithmeticException: / by zero

 at Fraccion.imprime(Fraccion.java:13)

 at Fraccion.main(Fraccion.java:56)

Java Result: 1

Haciendo manejo de excepciones hacer la corrección, en el método imprime, correspondiente para que no se genere el error y en lugar de eso manda un mensaje de que la fracción no es valida.

**Programación de Computadoras II**

**Tarea: 5**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

El máximo común divisor puede utilizarse para reducir una fracción, así por ejemplo, dada la fracción 63/84 el máximo común divisor mcd( 84,63) es 21. La fracción la podemos escribir como

(63/21)/(84/21) = 3/4

Escribir el objeto fracción con las siguientes características

|  |
| --- |
| **Fraccion** |
| - numerador : int - denominador : int |
| <<constructor>> Fraccion(unN: int, unD : int)- mcd() : int+ suma(unaFraccion: Fraccion) : Fraccion+ resta(unaFraccion: Fraccion) : Fraccion+ division(unaFraccion: Fraccion) : Fraccion+ multiplicacion(unaFraccion: Fraccion) : Fraccion+ imprime() : String |

El método mcd() se declara privado y será el responsable de calcula el máximo común divisor y se utilizará en el constructor de cada fracción, así por ejemplo cuando se declare

Fraccion a = new Fraccion(63,84);

System.out.println(a.imprime());

Se mostrará en pantalla 3/4

Los llamados a los métodos se hará de la siguiente forma

Fraccion a = new Fraccion(63,84);

Fraccion b = new Fraccion(45, 90);

Fraccion c = a.suma(b);

Fraccion d = a.resta(b);

Fraccion e = a.multiplica(b);

Fraccion f = a.division(b);

c.imprime();

d.imprime();

e.imprime();

f.imprime();

y cuando se ejecute el código se debe imprimir

5/4

1/4

3/8

3/2

 **Programación de Computadoras II**

**Tarea: 4**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Al dividir los números *a* entre *b* (números enteros), se obtiene un cociente *q* y un residuo *r*. Es posible demostrar que el máximo común divisor (mcd) de *a* y *b* es el mismo que el de *b* y *r*. Así pues podemos escribir

mcd(a,b) = mcd(b, r)

y

mcd(a, 0) = a

donde *r* es el residuo de la división entera de *a/b* y *a>b*.

Ejemplo. Calcular el mcd de 2366 y 273

|  |  |  |
| --- | --- | --- |
| Paso | Operación | Significado |
| 1 | 2366 dividido entre 273 es 8 y sobran 182 | mcd(2366,273)=mcd(273,182) |
| 2 | 273 dividido entre 182 es 1 y sobran 91 | mcd(273,182)= mcd(182,91) |
| 3 | 182 dividido entre 91 es 2 y sobra 0 | mcd(182,91)=mcd(91,0) |

Por lo tanto el máximo común divisor de 2366 y 273 es 91.

Implementar y reportar:

1.- El código Java con el método recursivo para calcular el mcd. El método se llamará mcd y recibe dos enteros y regresa el máximo común divisor,

2.- Una corrida con ejemplos del método funcionando y

3.- Conclusiones

**Programación de Computadoras II**

**Tarea: 3**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Dadas las secuencias de números

1. 3, 6, 9, 12, 15, 18, …
2. 5, 0, -5, -10, -15, …
3. 1, -1, 1, -1, 1, -1, …
4. 10, 9, 8, 6, 5,
5. 1, 2, 3, 5, 8, 13, 21, …

Escribir el código Java para generar dichas secuencias utilizando ciclos.

**Programación de Computadoras II**

**Tarea: 2**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Escribir la clase correspondiente al objeto Triangulo, el cual se puede describir por la siguiente tarjeta UML.

|  |
| --- |
| **Triangulo** |
| - a : float - b : float- c : float- A : float- B : float- C : float |
| <<constructor>> Triangulo(unA: double, unB : double, unC : double)+ Area() : float+ Perimetro() : float+ EsUnTriangulo(unA : float, unB : float, unC : float) : boolean+ info() : String |

donde a, b y c son los lados del Triangulo y A, B y C los ángulos entre los lados del triangulo. El método EsUnTriangulo prueba que los lados dados sean un triangulo factible y el método info() regresa toda la información relacionada con el triangulo.

Debes escribir una clase de prueba, llamada pruebaTriangulo donde hagas las definición y declaración de un objeto de este tipo. Debes probar con los siguientes triángulos

Triangulo T1 = new Triangulo(1,1,1);

Triangulo T2 = new Triangulo(3,4,5);

Triangulo T3 = new Triangulo(10, 11, 5)

Descripción

Poner aquí la descripción y detalles de lo que hiciste y como lo llevaste a cabo

Código implementado

Poner aquí el Código

Corrida

Dar una muestra de la corrida realizada

Conclusiones

Hacer un comentario y/o conclusión del trabajo desarrollado en la Tarea.

**Programación de Computadoras II**

**Tarea: 1**

**Nombre: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Escribir la clase correspondiente al objeto rectángulo el cual se puede describir por la siguiente tarjeta UML

|  |
| --- |
| **Rectangulo** |
| - base : double- altura : double |
| <<constructor>> Rectangulo(unaBase: double, unaAltura:double)+ Area() : double+ Perimetro() : double |

Debes escribir una clase de prueba donde hagas las definición y declaración de un objeto de este tipo.

Descripción

Poner aquí la descripción y detalles de lo que hiciste y como lo llevaste a cabo

Código implementado

Poner aquí el Código

Corrida

Dar una muestra de la corrida realizada

Conclusiones

Hacer un comentario y/o conclusión del trabajo desarrollado en la Tarea.